

**Assignment4报告**

**项目名称：gym**

**组 员 一：李 永 康**

**组 员 二：顾 堃 略**

# **1 立项报告**

# **1.1项目介绍**

### 1.1.1项目开发环境

基于AndroidStudio，运行平台为Windows。

# **1.2项目成员**

项目成员：李永康，顾堃略

# **1.3 项目特征**

|  |  |
| --- | --- |
| 项目类别 | 软件开发 |
| 项目名称 | 运动管理系统 |
| 团队规模 | 2人 |
| 难易度 | 普通 |
| 开发语言 | Java |
| 开发工具 | AndroidStudio |
| 操作系统 | Windows |

# **2 项目实现**

# **2.1项目具体实现流程：**

**运用sqlite3 进行数据存储，实现项目的脱机登录与离线浏览视频功能**

**（因功能类似，仅展示脱机登录部分）**

# OpenHelper类

public class OpenHelper extends SQLiteOpenHelper {

//建表语句

public static final String CREATE\_USER = "create table User ("

+ "username text, "

+ "userpwd text)";

public OpenHelper(Context context, String name, CursorFactory factory,

int version) {

super(context, name, factory, version);

// TODO Auto-generated constructor stub

}

@Override

public void onCreate(SQLiteDatabase db) {

// TODO Auto-generated method stub

db.execSQL(CREATE\_USER);//创建用户表

}

@Override

public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

// TODO Auto-generated method stub

}

}

public class SqliteDB {

/\*\*

\* 数据库名

\*/

public static final String DB\_NAME = "sqlite\_dbname";

/\*\*

\* 数据库版本

\*/

public static final int VERSION = 1;

private static SqliteDB sqliteDB;

private SQLiteDatabase db;

private SqliteDB(Context context) {

OpenHelper dbHelper = new OpenHelper(context, DB\_NAME, null, VERSION);

db = dbHelper.getWritableDatabase();

}

/\*\*

\* 获取SqliteDB实例

\* @param context

\*/

public synchronized static SqliteDB getInstance(Context context) {

if (sqliteDB == null) {

sqliteDB = new SqliteDB(context);

}

return sqliteDB;

}

/\*\*

\* 将User实例存储到数据库。

\*/

public int saveUser(User user) {

if (user != null) {

/\* ContentValues values = new ContentValues();

values.put("username", user.getName());

values.put("userpwd", user.getpw());

db.insert("User", null, values);\*/

Cursor cursor = db.rawQuery("select \* from User where username=?", new String[]{user.getName().toString()});

if (cursor.getCount() > 0) {

return -1;

} else {

try {

db.execSQL("insert into User(username,pwd) values(?,?) ", new String[]{user.getName().toString(), user.getpw().toString()});

} catch (Exception e) {

Log.d("错误", e.getMessage().toString());

}

return 1;

}

}

else {

return 0;

}

}

/\*\*

\* 从数据库读取User信息。

\*/

public List<User> loadUser() {

List<User> list = new ArrayList<User>();

Cursor cursor = db

.query("User", null, null, null, null, null, null);

if (cursor.moveToFirst()) {

do {

User user = new User();

user.setName(cursor.getString(cursor

.getColumnIndex("username")));

user.setpw(cursor.getString(cursor

.getColumnIndex("pwd")));

list.add(user);

} while (cursor.moveToNext());

}

return list;

}

public int Quer(String pwd,String name)

{

HashMap<String,String> hashmap=new HashMap<String,String>();

Cursor cursor =db.rawQuery("select \* from User where username=?", new String[]{name});

// hashmap.put("name",db.rawQuery("select \* from User where name=?",new String[]{name}).toString());

if (cursor.getCount()>0)

{

Cursor pwdcursor =db.rawQuery("select \* from User where pwd=? and username=?",new String[]{name,pwd});

if (pwdcursor.getCount()>0)

{

return 1;

}

else {

return -1;

}

}

else {

return 0;

}

}

}

跳转部分：

注册  
 public void onClick(View v) {

String name=et.getText().toString().trim();

String pass=et1.getText().toString().trim();

User user=new User();

user.setName(name);

user.setpw(pass);

Int result=SqliteDB.getInstance(getApplicationContext()).saveUser(user);

# 登录：

public void onClick(View v) {

String name=et.getText().toString().trim();

String pass=et2.getText().toString().trim();

//userList=SqliteDB.getInstance(getApplicationContext()).loadUser();

int result=SqliteDB.getInstance(getApplicationContext()).Quer(name,pass);

if (result==1)

{

state.setText("登录成功！");

}

else if (result==0){

state.setText("用户名不存在！");

}

else if(result==-1)

{

state.setText("密码错误！");

}